

POLIMORFISMO

3º Pilar da Programação Orientado a Objetos

O QUE É POLIMORFISMO ?

- Palavra originada do grego e significa "de muitas formas"
- Polimorfismo permite que objetos de diferentes classes possam ser tratados da mesma forma

IDEIA PRINCIPAL DO POLIMORFISMO

- No polimorfismo, podemos ter uma classe base e várias classes derivadas. A classe base define um conjunto de métodos que serão compartilhados pelas classes derivadas. No entanto, cada classe derivada pode implementar esses métodos de forma diferente, sendo elas por sobrecarga ou sobreescrita, .

EXEMPLO DE POLIMORFISMO

Classe Animal

metodo fazerSom()

Classe Gato estende Animal

metodo fazerSom()

Classe Cachorro estende Animal()

metodo fazerSom()

O compilador trata a classe gato e cachorro como se fosse a classe Animal, mas o método executado depende da classe que está sendo chamada, sendo possível apenas com herança ou interface.

POLIMORFISMO SOBRECARGA

Classe Somar:

metodo somar(inteiro a, inteiro b)

metodo somar(real a, real b)

metodo principal(){

 Somar soma;

 soma.somar(1, 1) -> 2

 soma.somar(1.5, 1.5) -> 3.0

No polimorfismo de sobrecarga ou estático, o compilador determina em tempo de execução qual método chamar, analisando os tipos e a quantidade de parametros, ou seja, a sobrecarga ocorre quando temos dois metodos com o mesmo nome mas com assinaturas diferentes, para decidir qual o método usar o java usa a regra do método mais especifico

POLIMORFISMO DE SOBRESCRITA

Classe Mae
 método falar()

O compilador sabe qual assinatura de método deve ser chamada, mas a implementação concreta só ocorre em tempo de execução de acordo com o tipo do objeto real

Classe Filha estende Mae
 // sobrescrita
 método falar()

Sabemos que x é do tipo Mae, então o compilador sabe que deve chamar o método falar() de Mae, mas em tempo de execução o objeto atribuído a ele foi a Filha, portanto o java chama Filha.falar()

Classe Principal
 método principal()
 Mae x = new Filho()
 x.falar() ?